



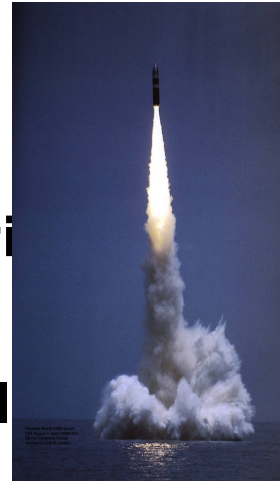
Y2K End-to-End Integration Testing and Contingency Planning Brief



Y2K Testing and Contingency Brief



- Assumptions and Guidelines
- Enterprise Testing
- Mission Critical Functions
- Functional Thin Threads
- Y2K End-to-End Integration Test Definition
- Contingency Plans
- System and Operational Contingency Plans
- Closing

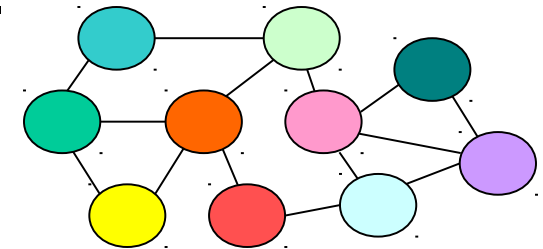




Unifying Themes



- **Follow the DoD Y2K Management Plan to ensure **Due Diligence****
- **Agreement on Terms & Definitions**
- **Interconnectivity -- the basis for End-to-End Testing and Contingency Planning**
- **Emphasize Sharing Information on Test Procedures and Contingency Planning**



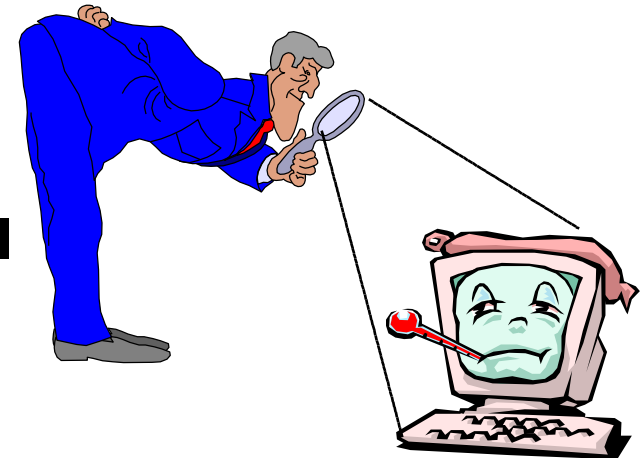


Unifying Themes



- Understanding that the expertise, resources, access and managerial focus needed to mitigate Y2K risk exist only at the process owner level
- Understanding how Y2K Managerial Responsibilities extend from the most senior to the most junior
- Understanding role of OASD (C3I) Y2K -

We have no “300-mile screwdriver”,
Sys know the systems
bes



“Y2K Bug”

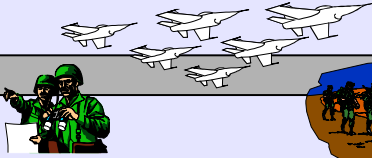


Enterprise Testing: Three Levels/Domain Focus



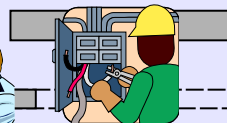
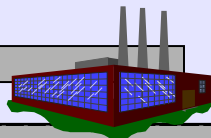
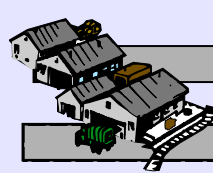
Defense-wide

**Mission-Centric
Tests**



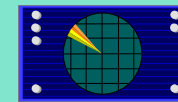
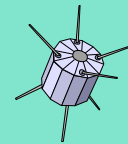
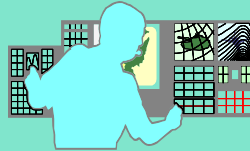
Warfighter Exercise Mission Execution Tests

**Functional-Centric
Tests**



Functional Systems Interface Tests

**Critical
Systems-Centric
Tests**



Individual System Tests

Services & Agencies



Mission Critical Systems Evaluation Proposal



Warfighting Missions

- C2
- Weapons
- Planning
- Nuclear
- Intel

CINCs

Warfighting Support Functions

- Personnel
- Finance
- Logistics
- Medical
- Communications

PSAs

Title 10/Infrastructure Systems

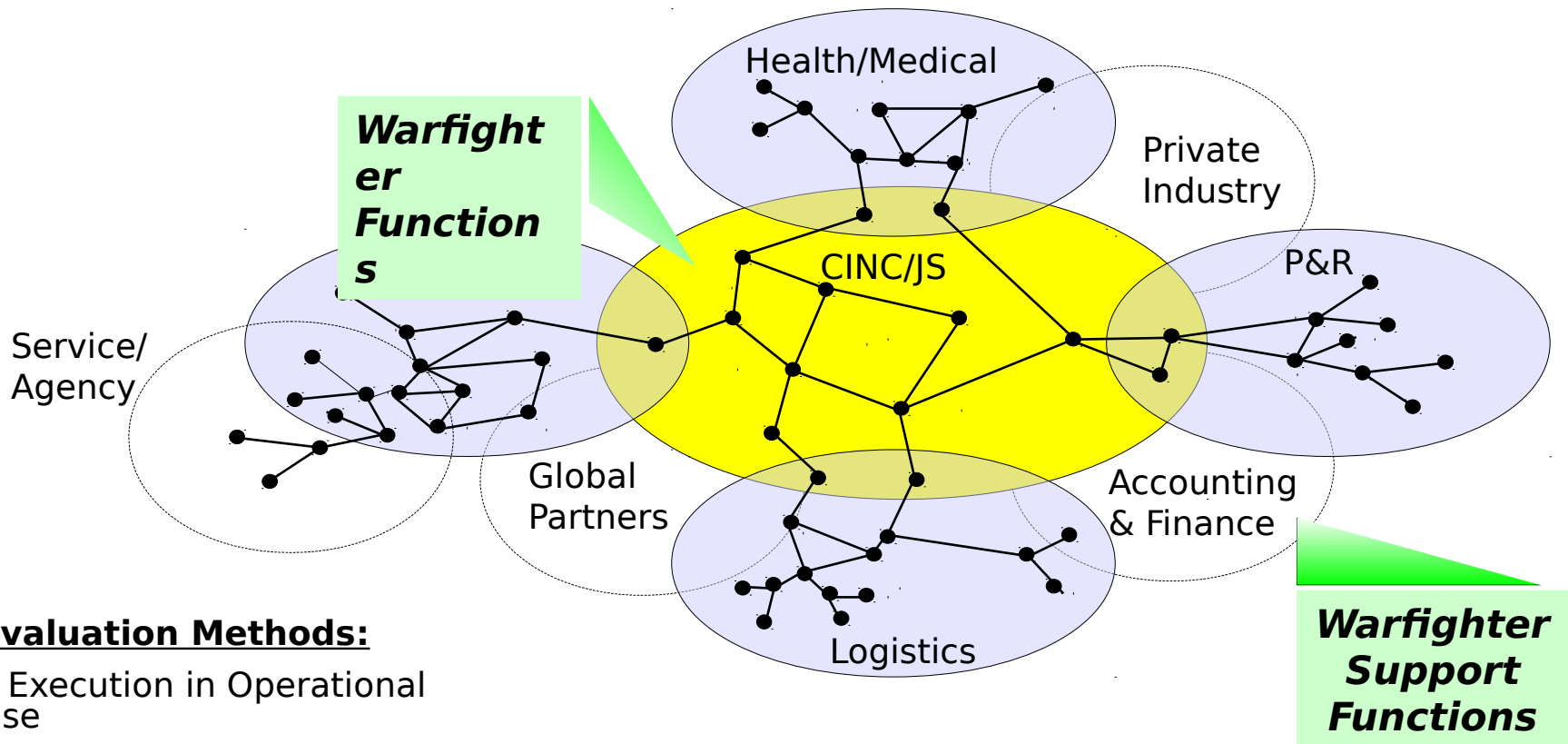
- Organize
- Train
- Equip
- Sustain

Services

Systems with no timing function



Integrated Evaluation Scheme



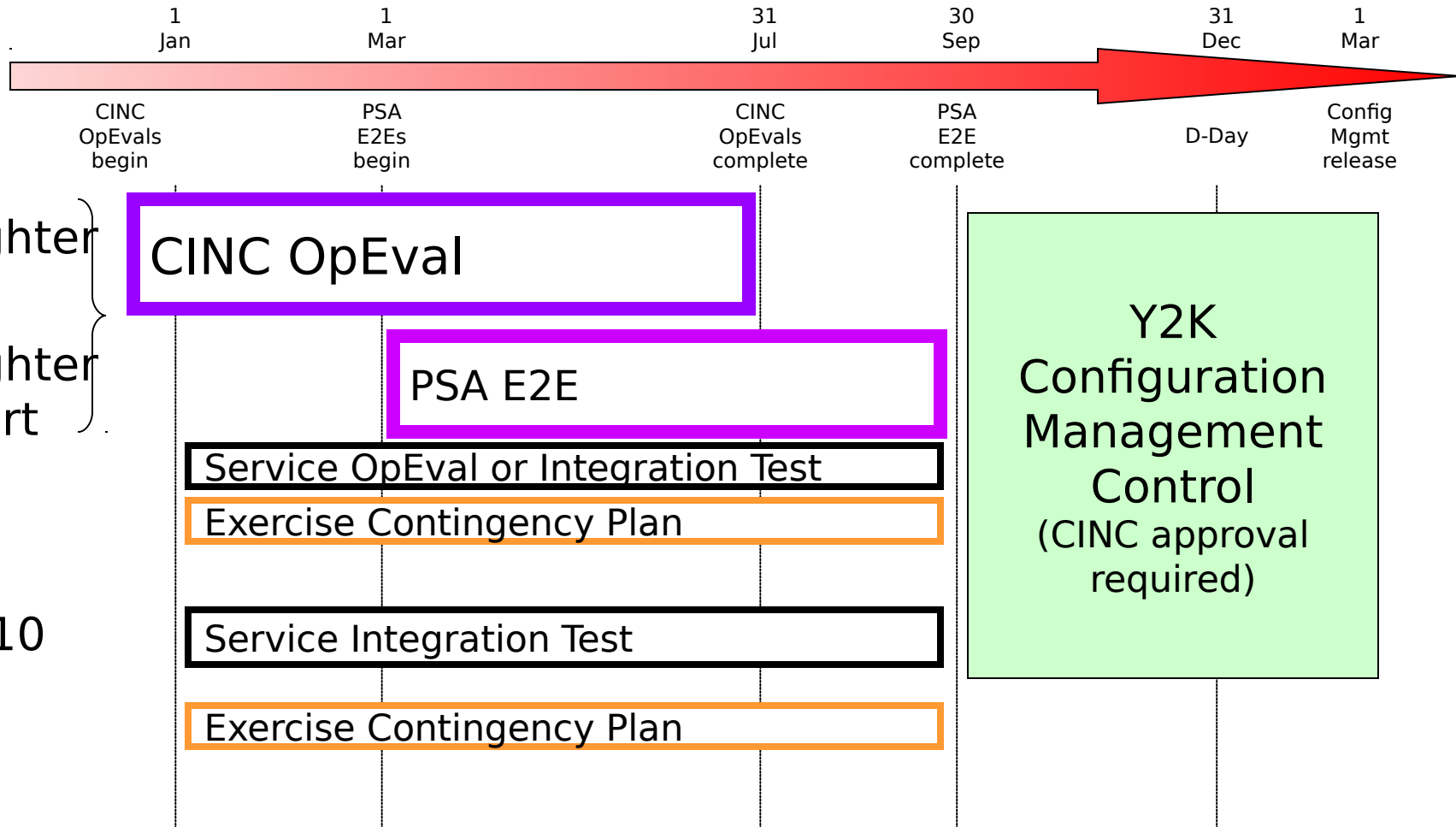
Evaluation Methods:

- Execution in Operational Use
- Operational Test Bed
- Federal Lab Tests
- Simulation/Modeling
- Inspections
- Contingency Plans
- CINC Plans identify mission critical systems and also those systems that present undue risk or are unfeasible
- OSD Y2K Staff will verify systems are included in Functional End-to-End evaluations or through other means



Year 2000 Evaluation Timeline

Mission Critical Systems





Y2K End-to-End Systems Integration Test



General Guidelines:

- All tests already conducted are “grandfathered”.
- These slides provide the Minimum general guidance, some systems may require more extensive testing.
- The system owners are responsible for **Due Diligence** during the testing process.
- Each service must coordinate testing requirements with the Joint Staff and CINCs.
- The assumption is that each individual system has already been completely analyzed and tested so that only the critical threads need to be exercised as part of the integration test.
- The goal of System Integration Tests is to ensure that each

orga
cri



ons



in



exerc



throu



r



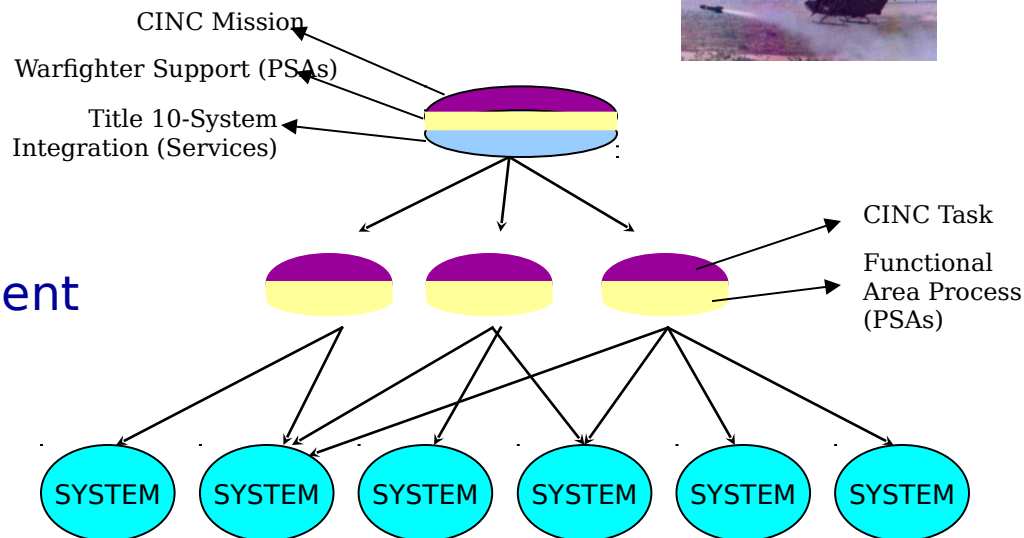
Y2K End-to-End Determine Mission Critical Functions



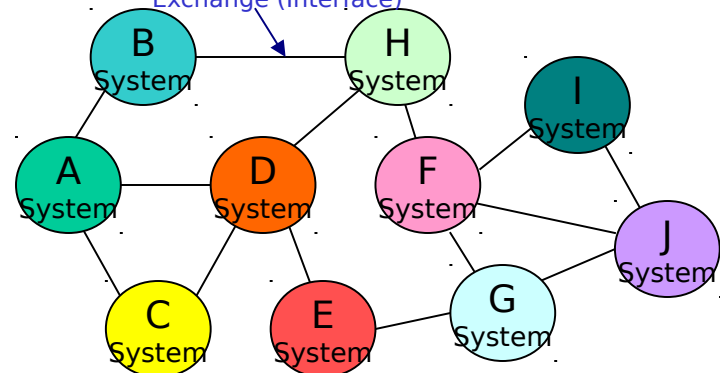
- Start with the organization's mission.
- Define the organization's functions.
- Determine the mission critical functions for the organization.

Example Functions:

- Crisis Planning
- ADP Support
- Force Deployment/Redeployment
- Readiness
- Sustainment
- Mobilization
- Intelligence
- Force Employment



System Information Exchange (Interface)



- Catalog the systems that support mission critical functions.
- Map the interaction and interfaces between the mission critical systems.

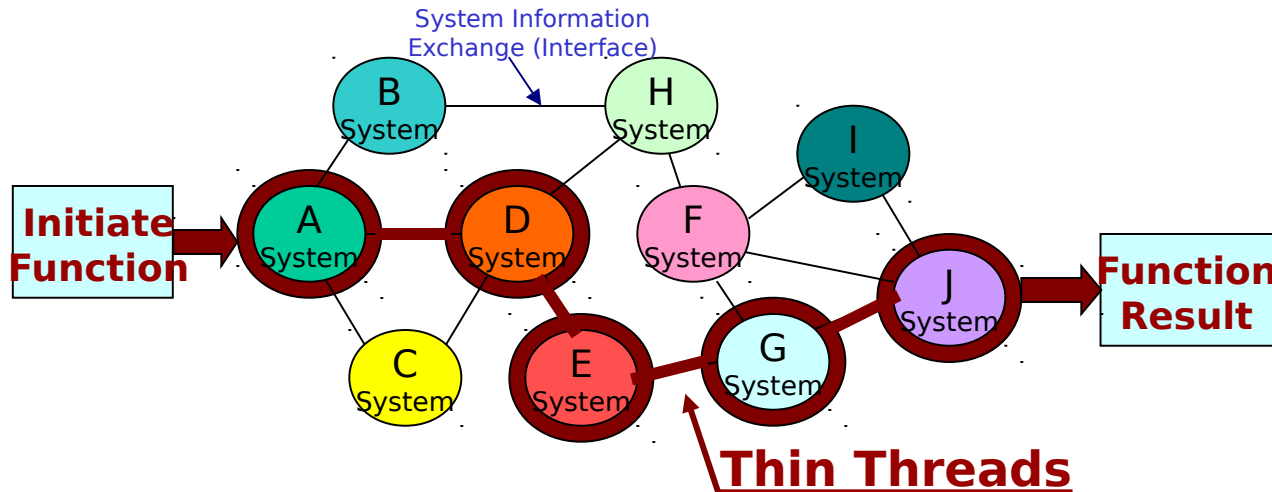


Y2K End-to-End



Define the Functional Thin Threads

- Develop the functional Thin Threads Strings.

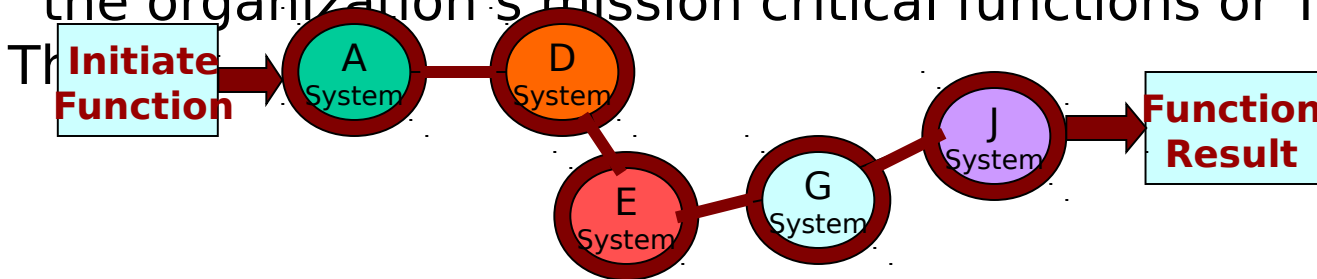


Examples of Functional Thin Threads:

- Sensor to Assessment to Target Acquisition to Fire
- Accession to Establish Base Record to Start Pay to Training to First Assignment

- Determine the Systems of Interest.
- Build an End-to-End Test Plan which exercises (at the minimum)

the organization's mission critical functions or Thin

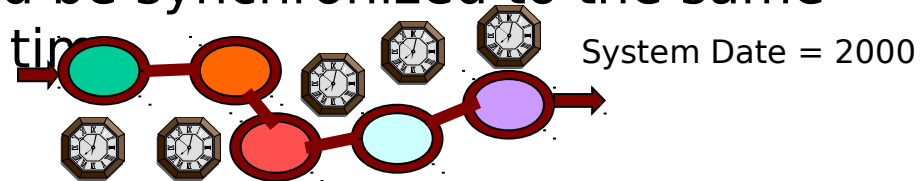




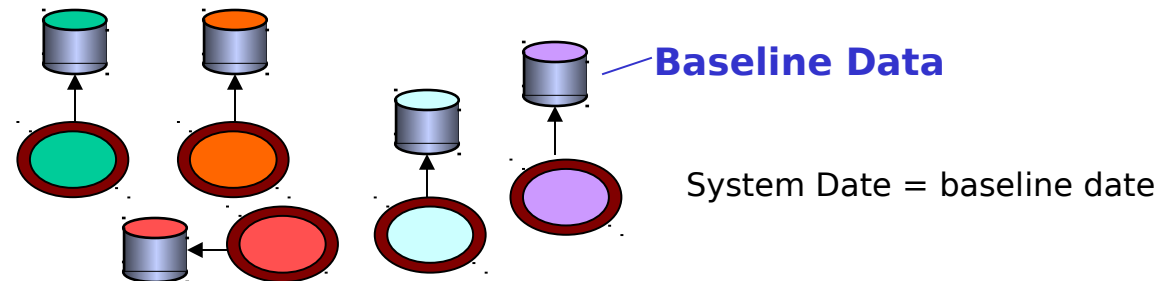
Y2K End-to-End Integration Test Definition



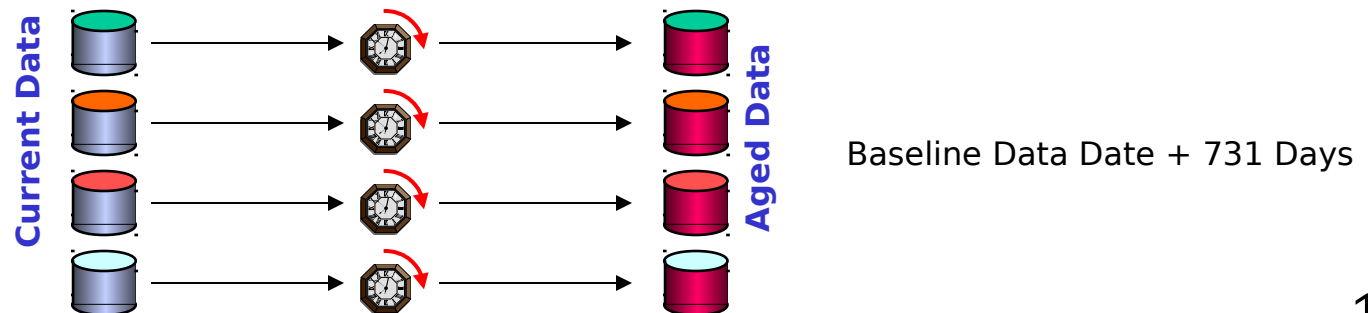
- In an ideal test, all systems would be synchronized to the same point in time, and interact in real time



- A current baseline of each system's data must be captured on the s
- The baseline (backup) must be stored for reuse during the integrati



- All baseline data files / databases must be aged to the same point in time in the future.



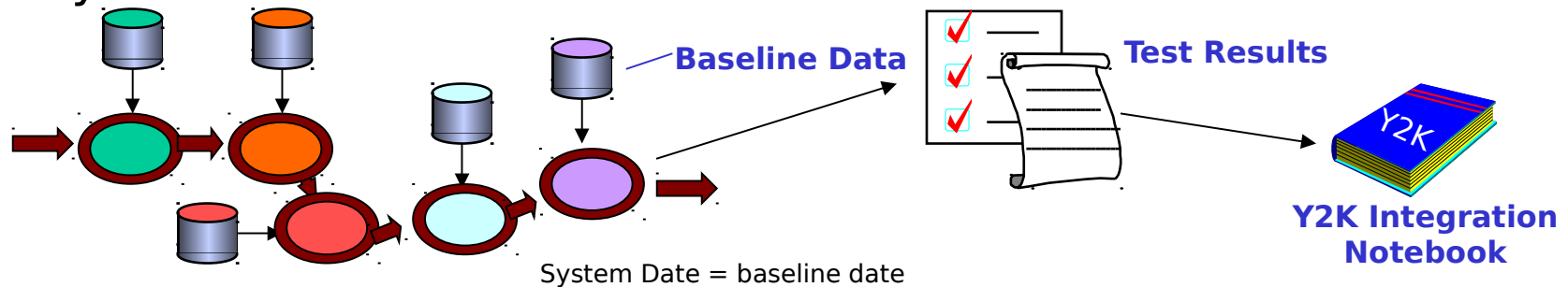


Y2K End-to-End

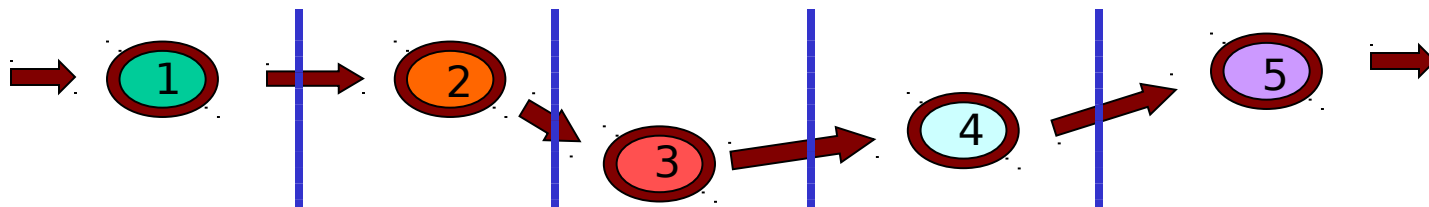
Integration Test Definition (continued)



- Integration Testing begins by running a baseline test using the base and system date. This validates the configuration and sets a test re baseline. Without this step there is no “control” test to validate the of any Y2K tests.



- All the actual electronic and physical outputs from each system must be passed to the next system in the functional Thin Thread of system integration.
- Note that all the systems have **synchronized data and system clocks** during each step of the Thin Thread test; however, all **system tests have to occur during a single test event.**



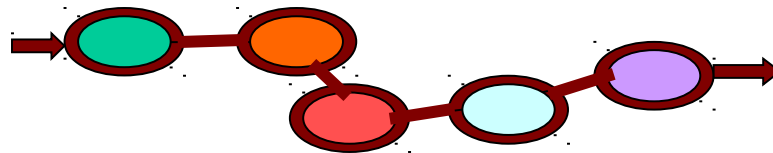


Y2K End-to-End

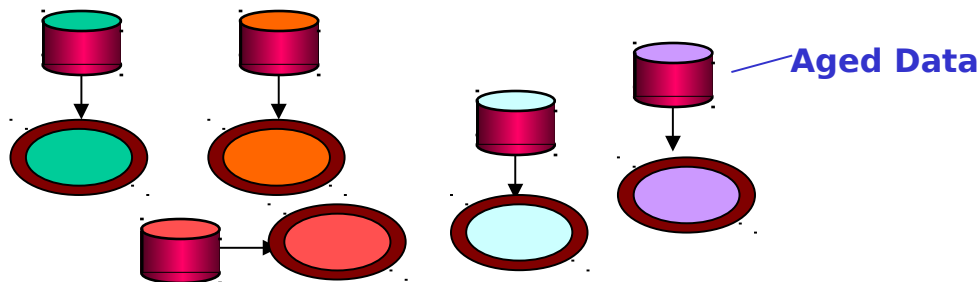
Integration Test Definition (continued)



- Note that only the mission critical functions needed to support the integration test are exercised within each system. Complete testing of each system should have been / will be completed as part of the individual system testing.



- Once the integration test baseline has been established, the system is re-tested in a Y2K environment. The same rules apply for the Y2K test as were applied during the baseline test.
- Begin by loading the aged data files and databases.



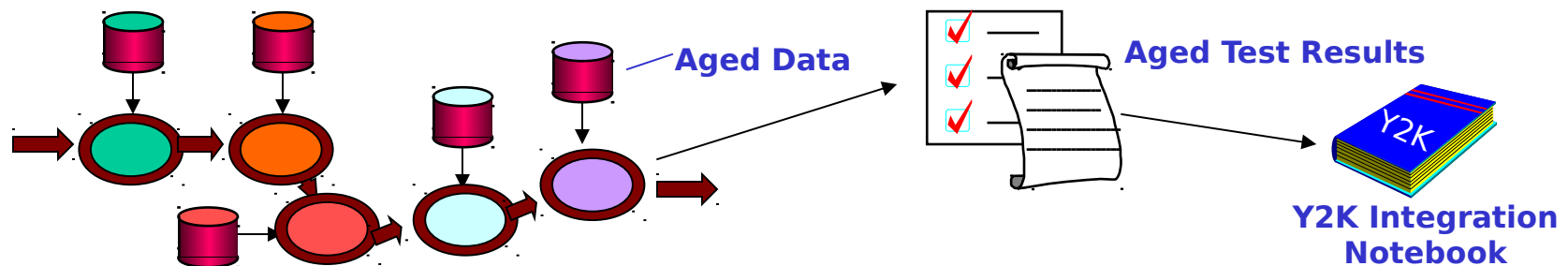


Y2K End-to-End

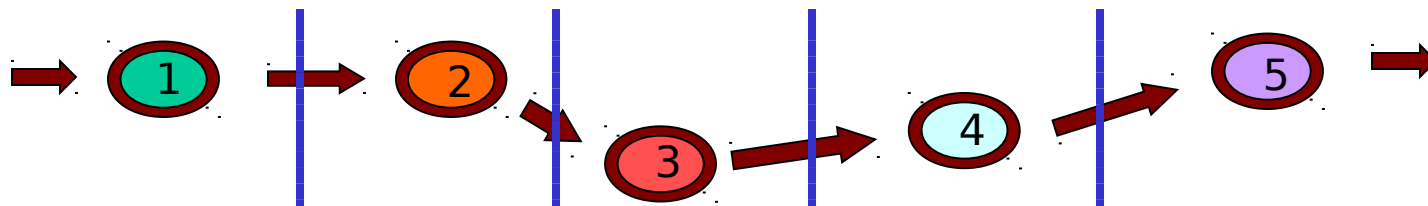
Integration Test Definition (continued)



- The system clocks need to be set to the same future date and time the test of each system with the Thin Thread.
- All the actual electronic and physical outputs from each system **must** be passed to the next system in the functional Thin Thread of system integration.
- As long as the data is aged to the same point in time, and interface data is saved so that it can be passed from one system to the next, the test can be conducted in stages.



System Date = Y2K dates, 12/1999, 1/2000, 2/2000, 10/2000, 12/2000, 1/2001



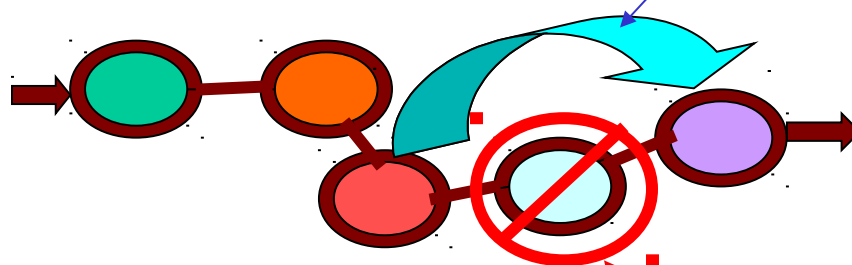


Y2K End-to-End



Incomplete Thin Thread Testing

- In the event that a “weak link” in a Thin Thread exists because a system in the Thin Thread has not yet been Y2K certified, end-to-end testing can continue by executing the contingency plan for the uncertified system. This allows end-to-end testing to proceed even when a Thin-Line string has not yet been completely certified.

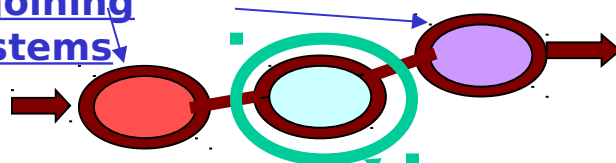


It is better to move forward so you know the integration results sooner, than wait for a 100% test that will occur too late.

System Not Yet Y2K Certified - “Weak Link”

- Once the “weak link” system has been certified. A follow on end-to-end may be conducted to prove interoperability with the adjoining system that at a minimum, the adjoining systems may be re-tested as part of follow on test.

Adjoining Systems

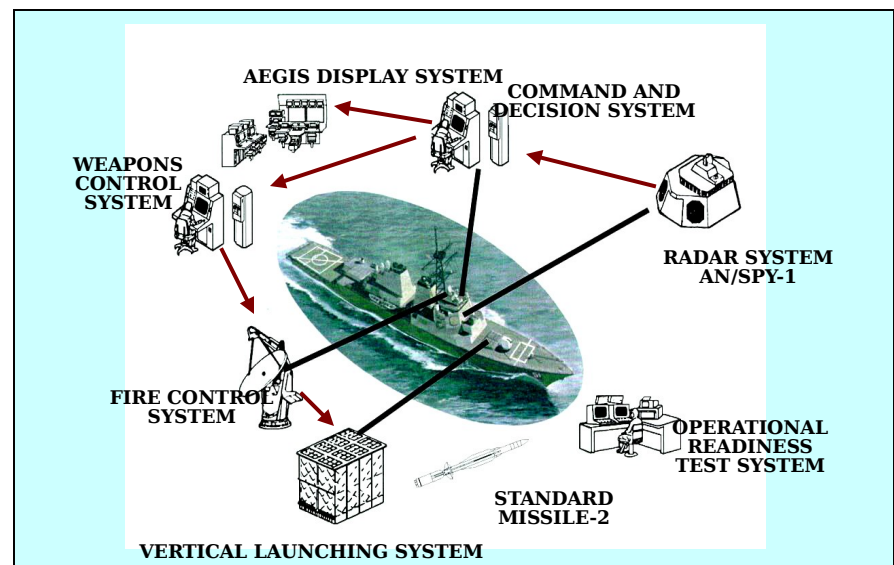
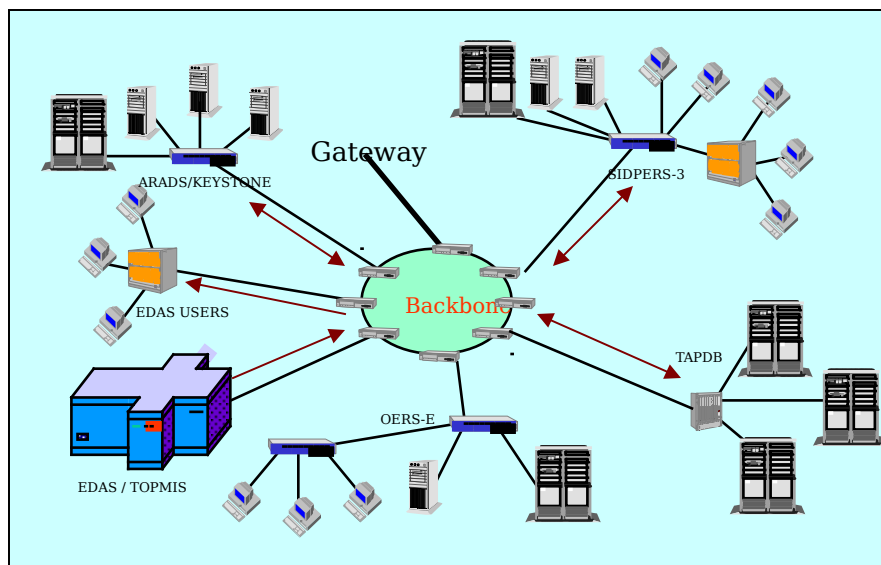
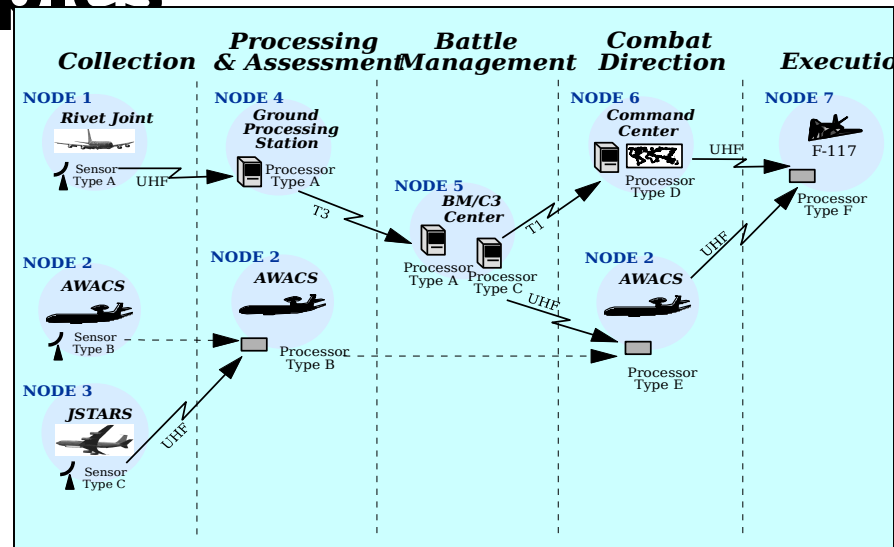
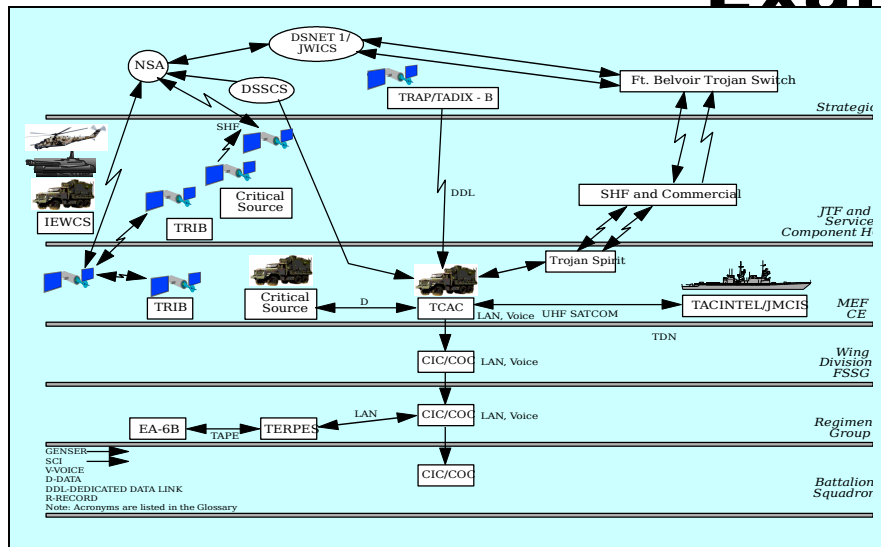


Follow on End-to-End Test

After “Weak Link” is Y2K certified



Y2K End-to-End Integration Test Thin Thread Examples



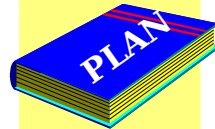
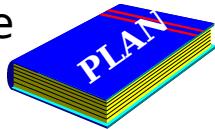


Contingency Plans



Assumptions:

- 3 Y2K Contingency Plans: **Program, System, Operational**
 - 2 Primary
 - Operational (Mission/Functional) CPs
 - System (Technical) CPs
 - 1 Additional
 - Program
- Program Contingency Plans With Interface MOUs Completed During Assessment Phase
- DoD 1999 Focus is on System and Operational Contingency Plans
- MC System Contingency Plans are Due by 12/30/98
- Operational Contingency Plans Complete by 03/31/99
- COOPs IAW DoDD 3020.26 Can Be Used as Y2K Operational CP **IF** Made Y2K Aware



Program [Fiscal]

Assumption: **Non Compliant System**
Responsibility: **PM**
Remediation/Renovation
Systems Under Development
Primary Emphasis
Cost
Schedule
Performance(Y2K)
Contains Alternative Strategies
COTS
Triage

Complete

System [Technical]

Assumption: **Compliant Systems**
Responsibility: **CIO**
Fielded Systems
Primary Emphasis
Restore Disrupted System
Contains 3 Levels of Concern
1. Health & Life Safety
2. Communication & Data Operations
3. General Operations

30 December 1998

Operational [Mission/Function]

Assumption: **System Not Available**
Responsibility: **CEO**
Primary Emphasis
Continue Mission or Function using

- Alternative Systems
- Manual Procedures/Processes

31 March 1999

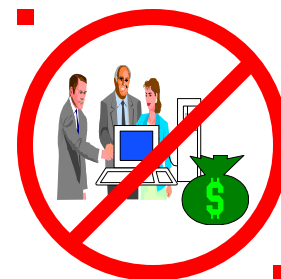


System Contingency Plans

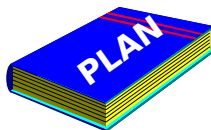


Assumptions:

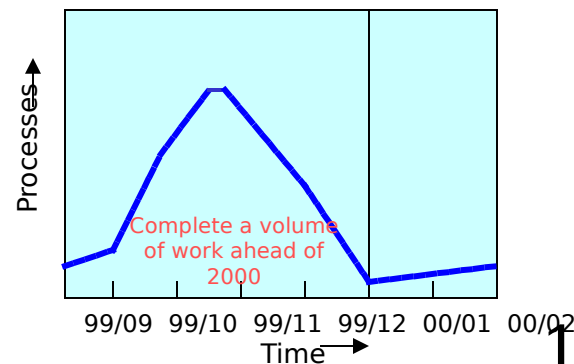
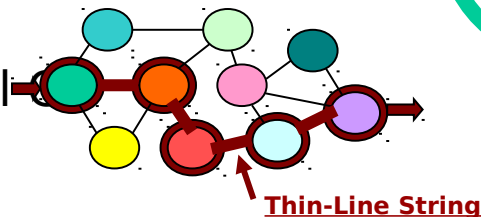
- IT/ITC/ITS Responsibility
- System is Y2K Compliant
- MC Systems Addressed In At Least One Operational CP
- Worst case scenario - Including:
 - No Additional Funding or Manpower Available.
 - Workstations and Office Automation Support May be Available.



Guidelines:



- System Specific, Tailored to Each Site
- Disruption Est. **>RTO** Triggers Operational
- Restore System Priorities
 - Health & Life Safety
 - Communication & Data Operations
 - General Operations
- Pre-position processes to avoid January 2000 window.
- Develop Matrix of All Critical Functions.
- Map Functions To Systems And Y2K Strategies.
- Define Y2K Strategies To The Task Level.





Operational Contingency Plans



Assumptions:

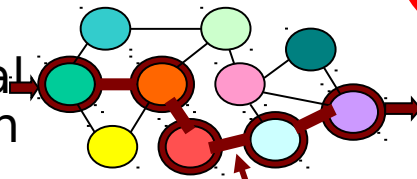
- Commanding Officer or Director Responsibility
- System(s) is/are Not Available
- Worst Case Scenario - Including
 - No Additional Funding or Manpower Available.
 - Extended Power and/or Communication Outages.
 - Operations Without HVAC or Water.
 - Some Facility Will be Available for Work Force.
 - Workstations and Office Automation Support May Not be Available.



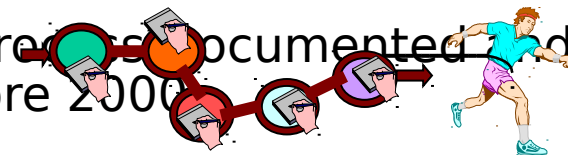
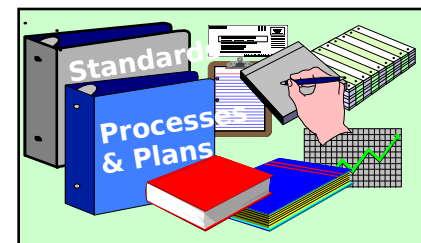
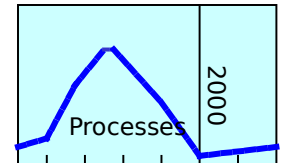
Guidelines:



- Operational CP Addresses all Mission Critical Systems in CO/Director's Functional Domain
- Y2K Aware COOP (IAW DoDD 3020.26) is Acceptable
- Critical Functions Must Continue to be Supported.
- Contingency Plan Based on Specific Function/Mission - Not Systems.
- Pre-position Processes to Avoid January 2000 Window.
- Have Manual Processes Documented in Place Well Before 2000



Thin-Line String

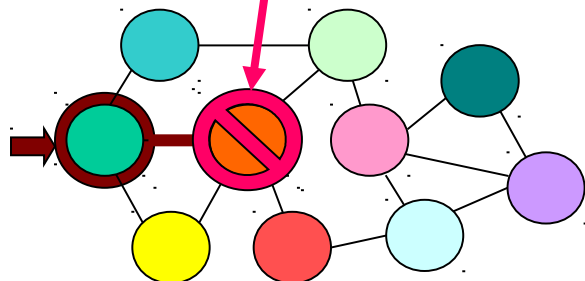




Y2K Contingency Plans



Y2K Disruption

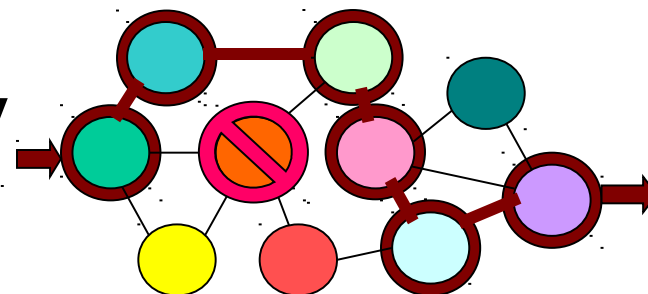


Note: System Disruptions Forecast to Exceed Recovery Time Objective (RTO), Triggers Disaster Recovery Plan

Activate System CP Immediately

Restore Disruption using Site Specific System Contingency Plan

- Prioritize Recovery
 1. Health & Life Safety
 2. Communication & Data Operations
 3. General Operations
- Restore Environmentals
 - * Power
 - * HVAC
 - * Water
 - * Telephone
- Generators
- Security Systems
- Fire Systems
- Elevators
- Move to Hot Site or Reciprocal Agreement Site if Required
- Recall Plan
- 1 800 Info/Status Recording
 - * Cellular Telephones
 - * Laptops
 - * Porta Potty
 - * Bottled Water
 - * Food
 - * Shelter



Operational CP

System Not Available, Utilize Alternative Systems or Manual Backup Procedures/Processes

Zero Day Strategy - Minimize Exposures & Reduce Backlogs in advance.

- No Personnel Transfers 9/99 - 3/00
- IT Personnel Full Time 12/29/99 - 1/4/00

Alternative Systems (F-117 Vs. Tomahawk)

1000# Charcoal

Pre-Printed Paychecks

Commercial SATCOM/Land Lines/ Cellular



Closing Points



- Follow the DoD Y2K Management Plan to ensure **Due Diligence**
- Use **Due Diligence** in Designing and conducting End-to-End Tests and in developing Contingency Plans
- Be a Good Manager - Find a way to **measure results**
- **Share information** with your stakeholders
- Review the Plans and Test Procedures of your critical suppliers





Background Slides



Defining an Individual System Certification Process



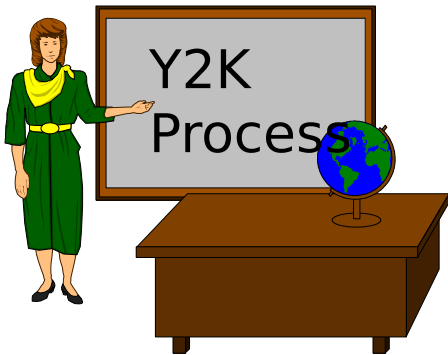
Year 2000 (Y2K) Application Software Certification Process

**Assumptions about
what
has been as part of
the
individual system
certifications**



Purpose of System Certification Example

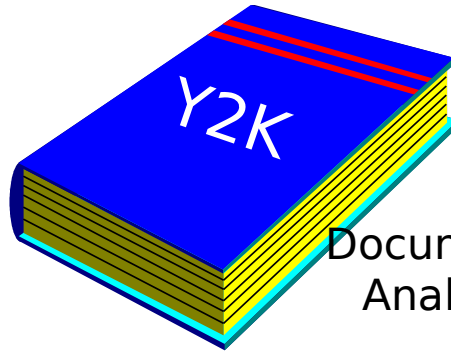
To provide an overview of DoD testing and validation process for Y2K certification.



Planning and Tracking



Functional User,
QA, and Software
Engineer Involvement



Documented Process,
Analysis, and Tests

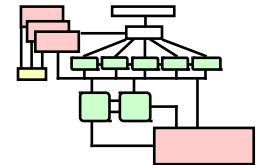
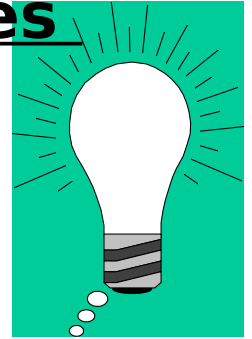


Year 2000 Organization Planning



In the 5 phase process, these are examples of the actions performed:

- Inventory all applications
- Establish "Health" of Code - "Red" or "Yellow"
- Derive a Level of Effort Metric - How long to fix?
- Establish application priorities
 - Critical to organization
 - Internal Interdependencies
- Develop and document universal processes to:
 - Assess Code
 - Fix Code
 - Test Code
 - Certify Code





Year 2000 Organization Planning



Y2K Project Planning

- Brief System Level CCBs to:
 - Obtain approval on project schedules
 - Provide impact analysis re: Non-Y2K Projects
 - Identify coordinating functional representatives



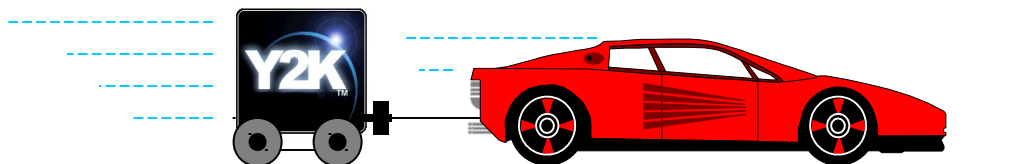
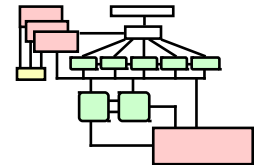
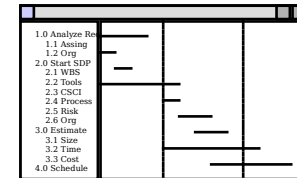


Year 2000 Organization Planning



Y2K Project Planning

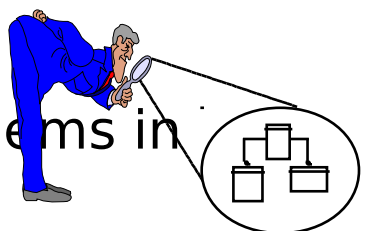



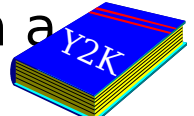

- Develop Project Schedules
 - Allocate and Manage Scarce Resources
 - Develop Impact and Risk Analyses - Identify Shortfalls
 - Manage Each Application through to Certification
- Conduct a Pilot Project to Validate Processes
 - Pilot Projects
 - Validate our Processes and Level of Effort Metric
 - Formal Kick-off of Y2K Project Plan
- Run a Y2K Test Drive of “Critical” Applications





Certification Process Overview



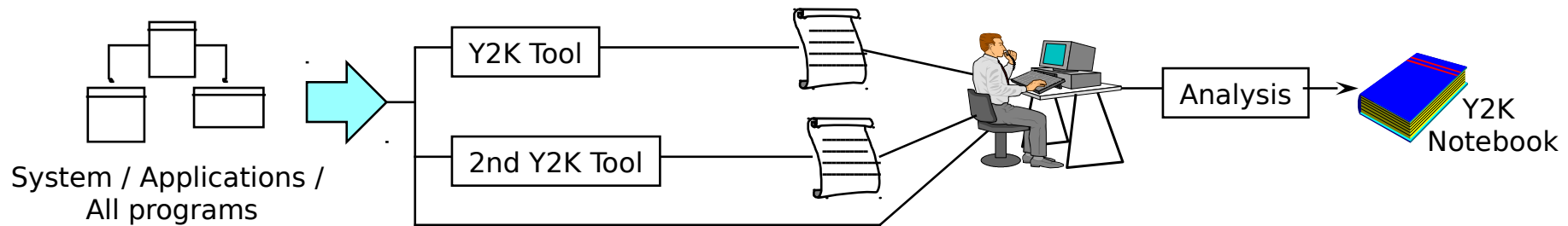
- **Analyze** - Look for any Y2K processing problems in all code in every system. 
- **Correct Any Y2K Problems Found** - Expand dates or use sliding windows. 
- **Create Test Plans** - Plan to test every date related function in every system. 
- **Test All Parts of Each System** - With QA and the functional users witnessing the tests, all date processing is tested in a Y2K environment. 
- **Document All Test Results** - Hard copies are kept in a notebook to document every test. 
- **Certify System** - The functional users, system managers, and 



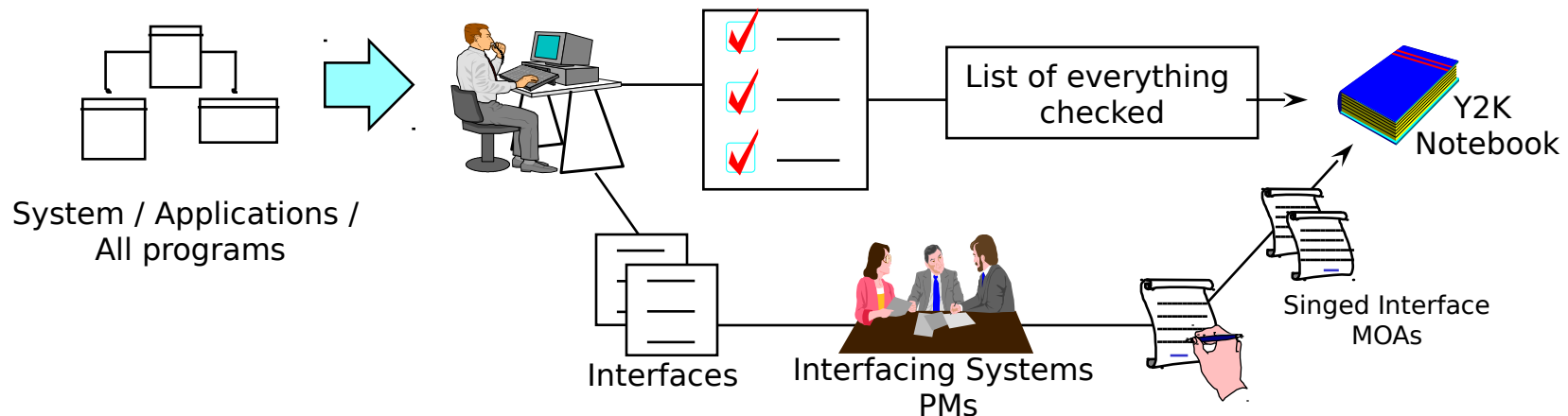
Certification Process



- All dates are identified by using a Y2K automated tool and manual analysis.



- Each programmer uses a standard checklist to help identify any date related problems during the analysis of the code. Interfaces and MOAs are defined and signed with all interfacing systems/PMs



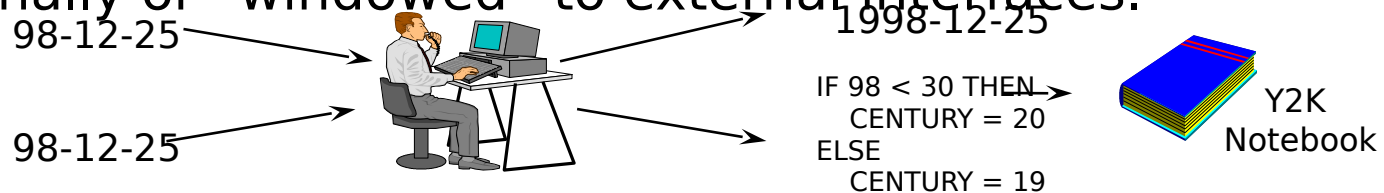


Certification Process



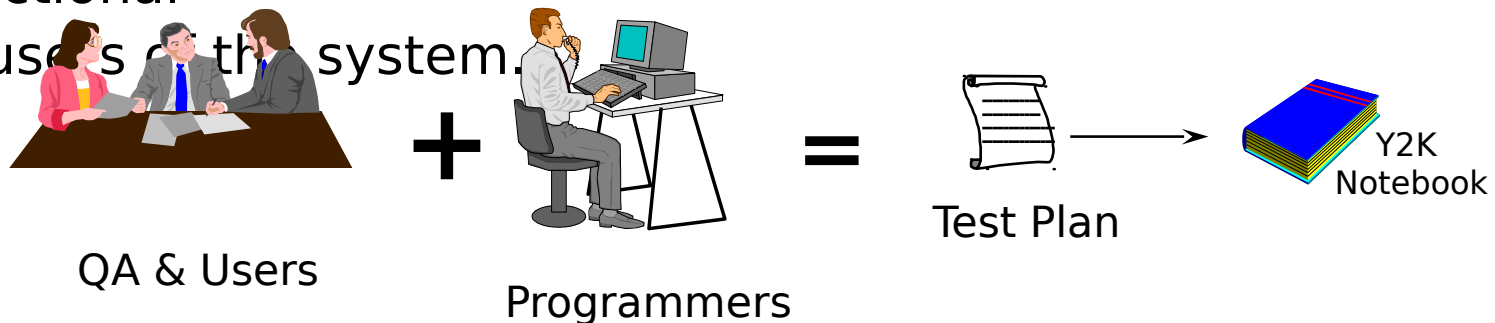
- If the system uses a two character year then date fields are expanded

internally or “windowed” to external interfaces.



- Test plans which identify all date-related operations are prepared by a

team consisting of the programmers, V&V (or IV&V) and the functional users of the system.



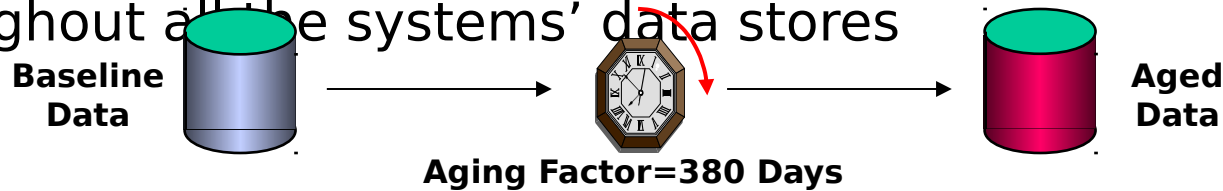


Certification Process



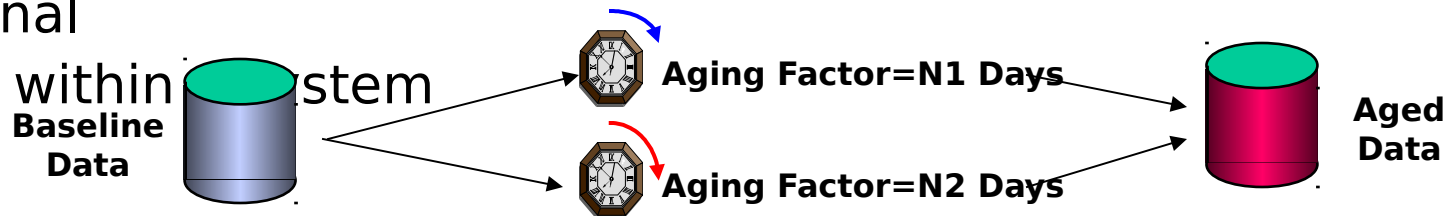
- In many systems, dates can often be rolled forward a consistent amount

throughout all the systems' data stores



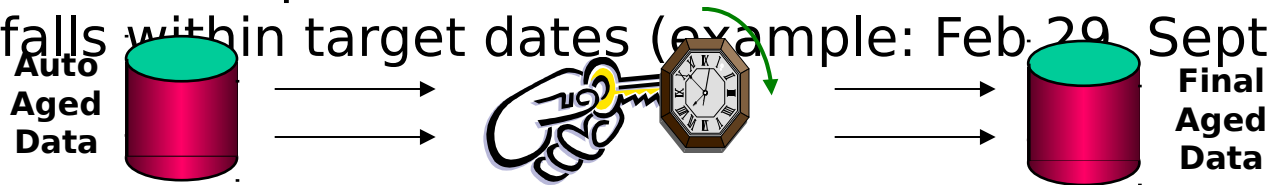
- Some systems may require specific rules for aging different functional

dates within a system



- Often manual aging must be done to ensure that a representative sample

of data falls within target dates (example: Feb. 29, Sept 30/Oct 1st)



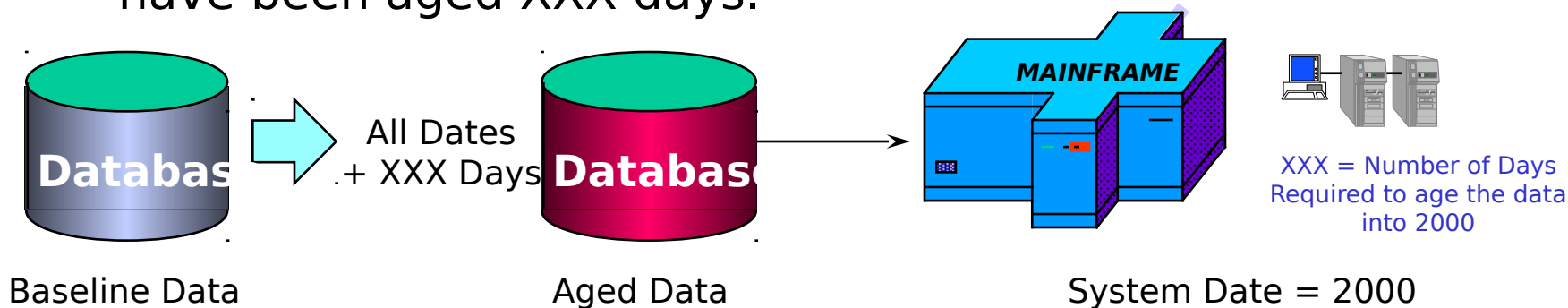
Note: Data aging is required to accurately test any information system that stores date related information in files or databases. Without an appropriately aged database it is possible that sort routines and keying functions may not be adequately tested when validating the application.



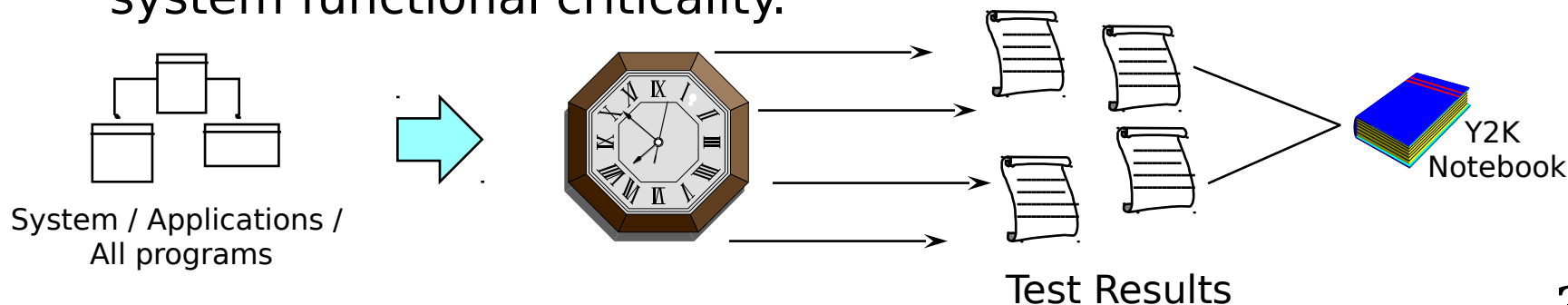
Certification Process



- A separate environment (LPAR or Y2K servers) is set up which allows the date to be set for each test. All date fields on every part of the database and files have been aged XXX days.



- Every program and subprogram is tested 6 times using these scenarios: current date(baseline); 31 Dec 1999; 1 Jan 2000; 29 2000, Oct 2000, Jan 2001. Number of dates tested depends on system functional criticality.



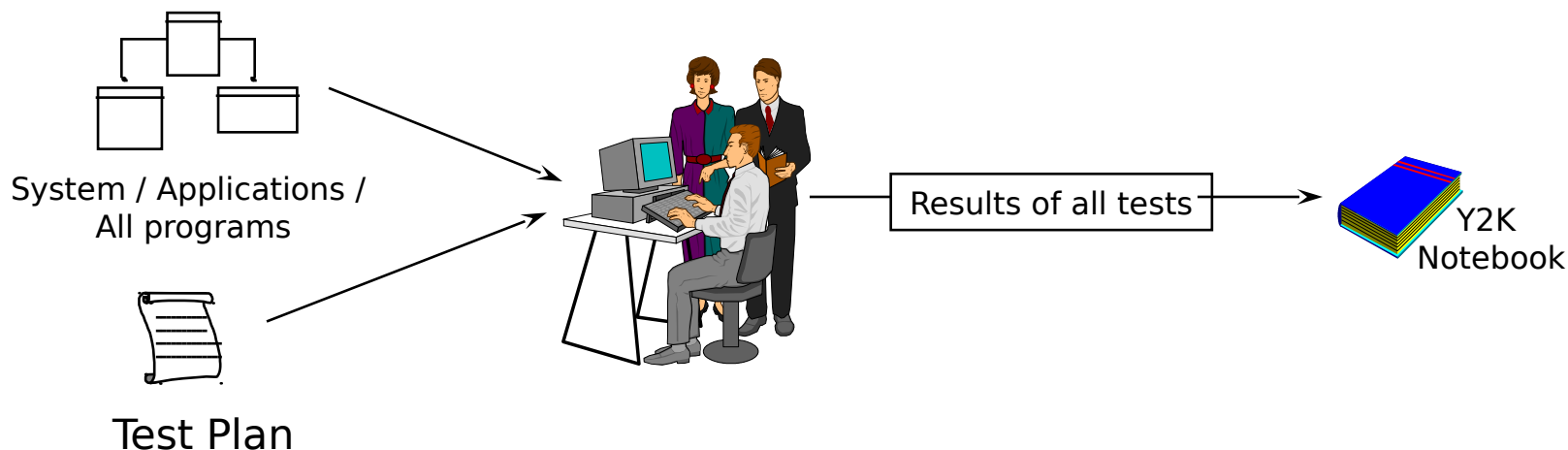
Test Results



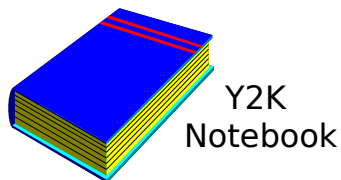
Certification Process



- Y2K test is executed with programmer, functional and QA together.



- A notebook is prepared for each application containing the test test results, and functional approval (functional & QA personally the tests and review screen prints, report output, file dumps).

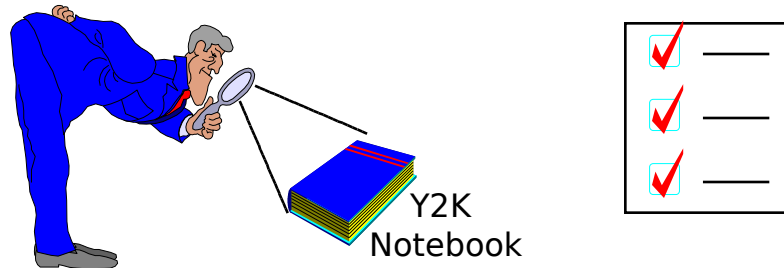




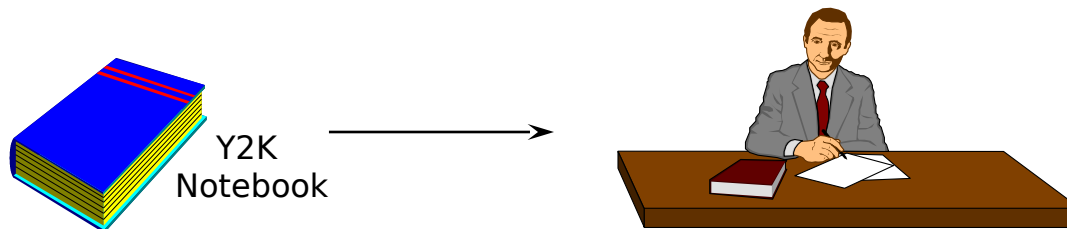
Certification Process



- Test results and notebooks validated independently by V&V (or IV&V).



- **Notebooks certified** - critical systems by CG; by-name systems by Chief of Staff; non-critical systems by CCB chair or Director.

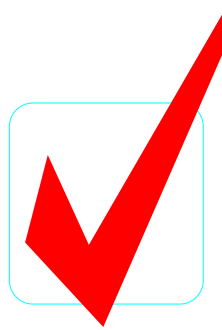
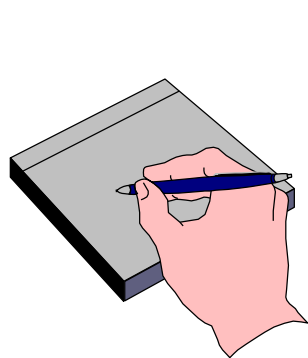




Certification Process



- What does Certification mean?
 - A formal signature stating that an application is Y2K certified documents that due diligence occurred during the five phases of awareness, assessment, renovation, validation and implementation in accordance with the DoD Y2K Management Plan and Compliance Checklist.

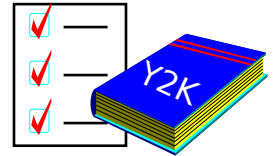
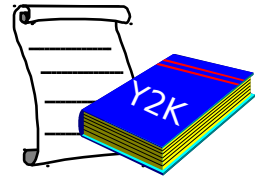




Defining Software Independent Validation and Verification (IV&V)



- **Independence:** Applied with managerial and/or financial independence.
 - Contractor, other than developer
 - Government, other than developer
 - DoD Independent Lab such as JPLC
- **Verification:** The process of determining whether or not the products of a given phase of the software development cycle fulfill the requirements established during the previous phase.*
 - Accomplished for each step in the development effort
 - Verifying entry and exit criteria for each Y2K remediation phase
 - Verifying evidence of due diligence
- **Validation:** The process of evaluating software at the end of the software development process to ensure compliance with the software requirements.*
 - Final system testing and certification
 - Used to verify that the system performs as intended in its entirety after Y2K remediation efforts have been completed
- **Full Lifecycle IV&V:** Dependent on the complexity, criticality, and required integrity of the system.



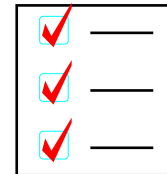
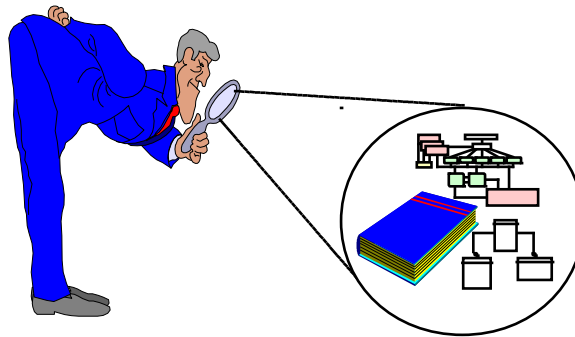


Typical IV&V Activities



- Requirements Analysis
- Development Analysis
- Code Analysis*
- On site Audits
- Document Inspection
- Algorithm Analysis
- Training Evaluation
- Milestone Review
- Test Planning*
- Test Witnessing*
- Independent Testing*
- Data Analysis*

*Typically applied to Y2K remediation/validation efforts.



Software, Documentation,
and Process



Certification Levels



LEVEL

DESCRIPTION

0

System retired or replaced

1

Full independent testing

1a

Full independent testing completed using a four digit year format

1b

Full independent testing completed using a two digit year format

2

Independent audit

2a

Independent audit of system and existing testing completed using a four digit year format

2b

Independent audit of system and existing testing completed using a two digit year format

3

Self-Certification

3a

Self-certification with full use of 4 digit century date fields

3b

Self-certification indicates risk due to use of 2 digit century fields

3c

Self-certification indicates risk due to ambiguous usage of dates

3d

Self-certification indicates potential problems

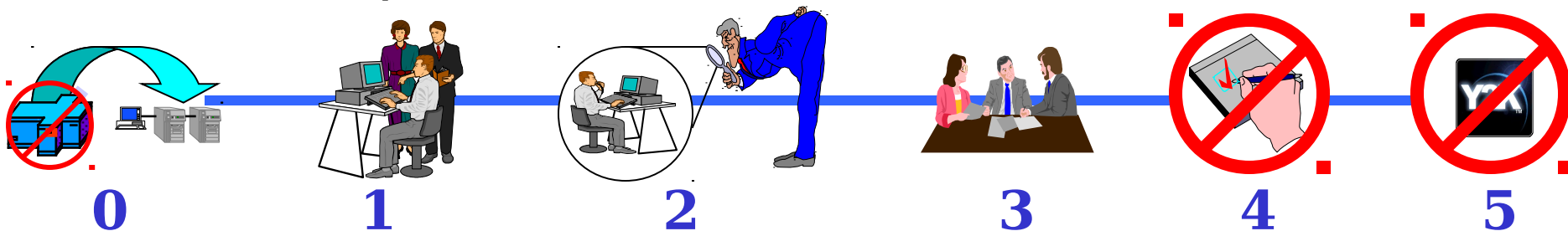
(System needs additional work before Year 2000 processing can be assured with any level of reliability)

4

Not certified or system requires additional work.

5

Does not process date related data



0

1

2

3

4

5



Independent and Audit For Levels 1 and 2



If the system development, remediation and/or testing is performed by a:

Then the independent testing or Auditing may be performed by a:

Contractor

- ✓ • Government Organization
- ✓ • Different Contractor
- ✓ • Independent Test Facility, e.g., JITC, etc.
- ✓ • Inspector General (IG) Office
- ✓ • Independent Audit activity



Government Organization

- ✓ • Contractor
- ✓ • Different Contractor Organization not involved in development / remediation / testing
- ✓ • Independent Test Facility, e.g., JITC, etc.
- ✓ • Inspector General (IG) Office
- ✓ • Independent Audit activity

Note:

1. Certification effort must be independent of system development or maintenance tasking
2. Certification tasking must be to a technically competent and knowledgeable.

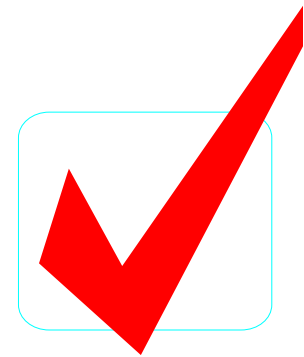
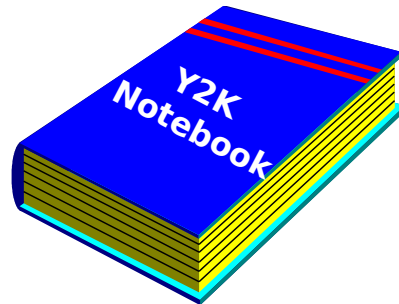


Maintenance of Y2K Certification



The following criteria must be met to ensure certified software retains certification:

- Recommend a Freeze to changes on systems that have been Certified
- Software must be under a Software Configuration Management Plan.
- A software change tracking system (e.g. ECP) must be used.
- All software changes to certified systems must under go Independent Validation and Verification (IV&V).
- Degree of IV&V will depend on type of change.
- Each change must have a corresponding addendum to the system's Y2K notebook.





Maintenance of Y2K Certification



<u>Date Category</u>	<u>Minimum Testing Level</u>	<u>Minimum Documentation Delivered to IV&V Unit</u>
Software Change did not involve date variable or parameter.	<ul style="list-style-type: none">• Use accepted testing practices	<ul style="list-style-type: none">• Current established documentation• Documentation must bear Category Designation
Software Change involved date variable or date parameter used in display or output.	<ul style="list-style-type: none">• Use accepted testing practices• Self Test using aged data for 29 February 2000	<ul style="list-style-type: none">• Current established documentation• Documentation must bear Category Designation• Test Plans, test results and applicable checklist(s) for Self Test
Software Change involved date variable or date parameter used in logic branch, calculation or data query	<ul style="list-style-type: none">• Use accepted testing practices• Integrate Team Test using aged data for 29 February 2000	<ul style="list-style-type: none">• Current established documentation• Documentation must bear Category Designation• Test Plans, test results and applicable checklist(s) for Integrate Team Test



DEV SELF-TEST

For applications following the Self-Test path, the Developer will prepare a Test Plan and conduct a pass using data aged for 29 February. After evaluating the test results and assembling their complete Application Notebook addendum (test Plan, copies of input, copies of output, etc), Developers will present their findings to IV&V unit.

INTEGRATION TEAM TEST

Integration Team Tests are required for all Software Changes involving date variable or date parameter used in logic branch, calculation or data query. When all of the modules in an application requiring date changes have been tested in the normal development area and the application is ready for movement to Production, the Branch Chief will request testing participation from the Y2K Coordinator. Joint Teams composed of Developers, Functionals and IV&V will then be assigned to conduct test using data aged for 29 February 2000. This will ensure that all modified programs will function correctly in the year 2000. After evaluating the test results and assembling their complete Application Notebook addendum (test cases, test results, ECP, and metrics), the RMA will be presented to QA.